Enterprise Application Development in the Cloud Workshop - Architecture Overview

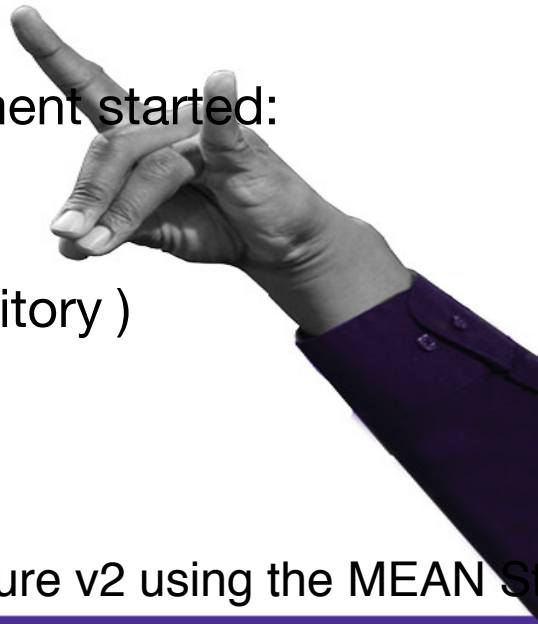
v0.1

Introduction

- The next generation Development Platform for developing Enterprise Applications will be browser and cloud based.
- This Workshop will demonstrate what this Development Platform will look like and give students a hands on opportunity to experience this platform.
- The Development Platform will consist of the following components:
  - ★ Cloud based IDE - Codenvy
  - ★ Cloud based Development Runtimes - Codenvy
  - ★ Cloud based Production Runtimes - Redhat OpenShift (on PaaS)
  - ★ Cloud based Source Control System - Github
  - ★ Cloud based Automated Build System - Redhat Openshift (using Jenkins)
  - ★ Cloud based DevOps Automation - Jenkins, JUnit, Maven, JMeter (using flood.io)

Reference Architecture and SDK

- The Workshop will leverage a Reference Architecture as a teaching tool:
  - ★ Fully functioning end to end system (using one IoT application and two enterprise applications)
  - ★ Applications written in Java and PHP (languages the students already know)
  - ★ Fully documented (providing an example of engineering rigor and discipline used to build a SDK)
  - ★ Will be used as guide and for a reference within the class materials
  - ★ Will provide the students with all the scaffolding to get their Cloud development started:
    - ✓ Spring Frame application template (can be cloned from a GIT repository)
    - ✓ Laravel Application application template (can be cloned from a GIT repository)
    - ✓ Requirements and Documentation
    - ✓ Step by step instructions for how to get started



NOTE: depending on the interest I am also thinking of building a Reference Architecture v2 using the MEAN St

Reference Applications

- The following applications will be built in the Workshop using the Cloud based Development Platform:
  - ★ IoT Services App - Using Tomcat written in Java leveraging JAX-RS, JSON, and the Spring Framework
  - ★ IoT Reporting App - Responsive browser based written in PHP leveraging Bootrap (for responsive support), Guzzle (for HTTP Client for REST API), LavaCharts (for charting), and the Laravel Framework
    - ✳ An IoT App using a Raspberry Pi 3 written in Python consuming a REST API published on the IoT Server will be available for use in the Development Platform. The IoT application will be developed on the standard Raspberry Pi development environment outside of the Workshop.

IoT Logical Architecture

IoT Application
(Sense HAT)

IoT Application
(Raspberry Pi)

REST API

HTTPS
HTTP Basic Auth.
Username/Password Auth.

laravel

php V7

IoT Reporting Application

REST API

HTTPS
HTTP Basic Auth.
Username/Password Auth.

Java POWERED

spring

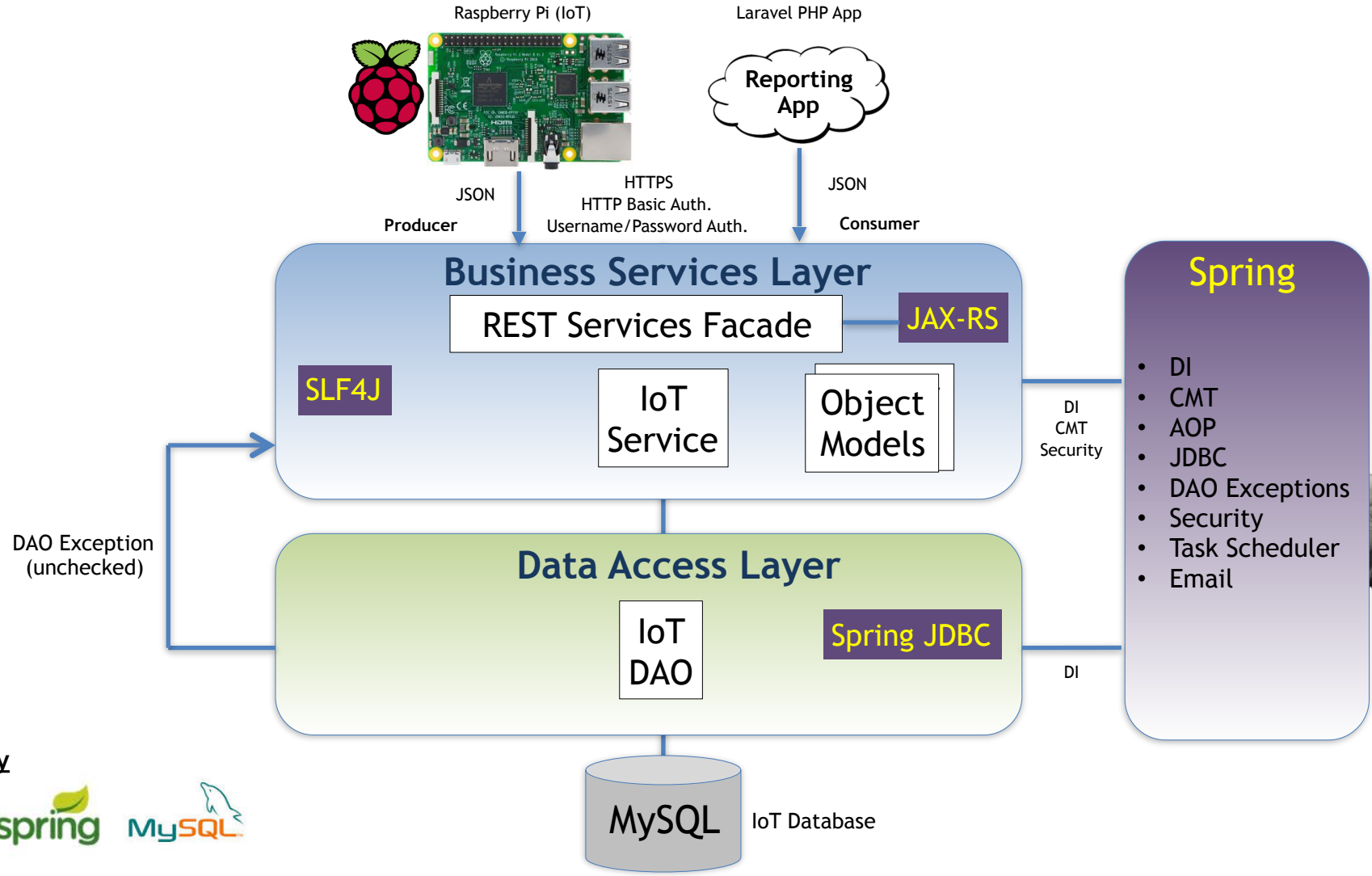
MySQL

IoT Server Application

SDK

- Reference Apps
- Developer Templ.
- Docs
- Tools

IoT Services Application Logical Architecture



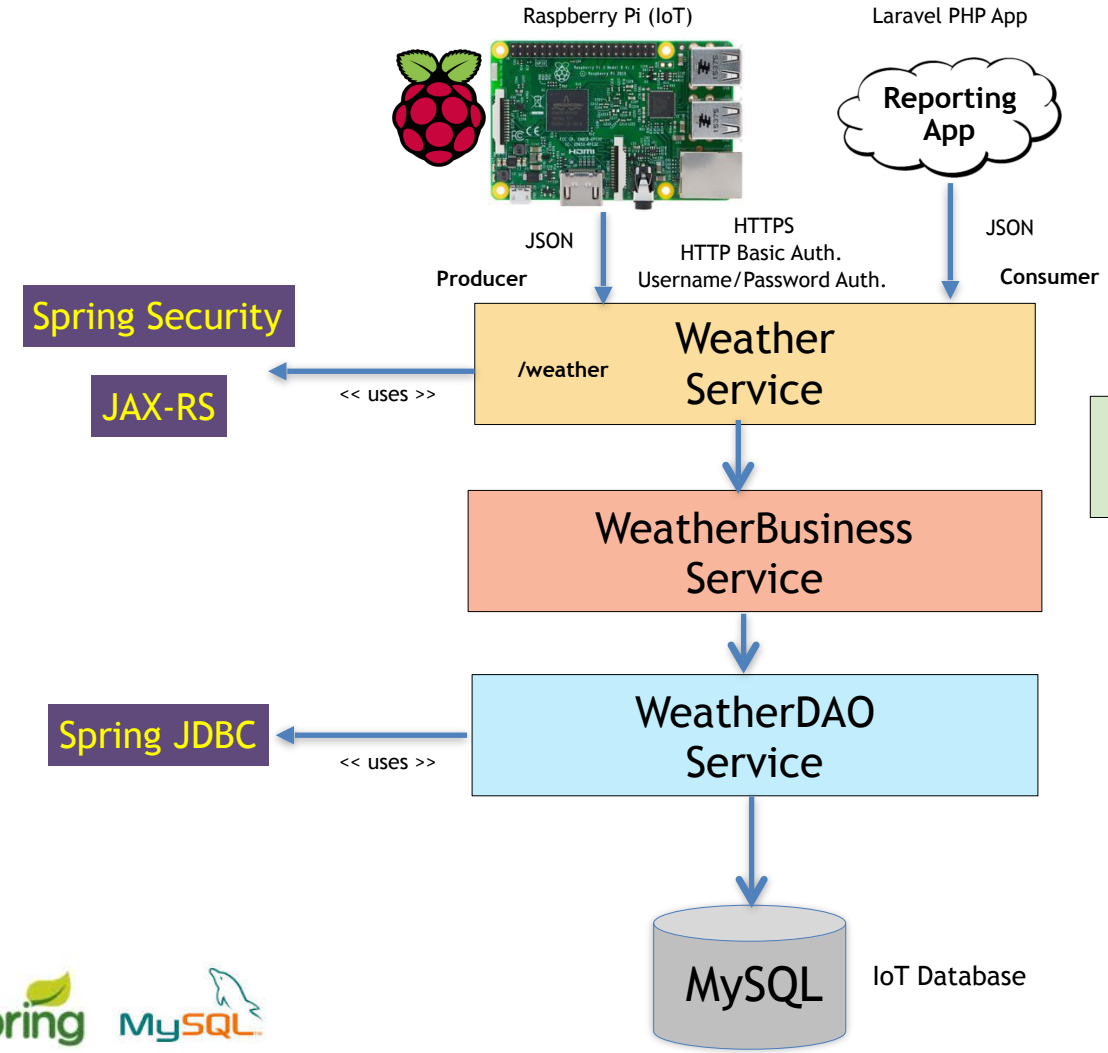
Powered By

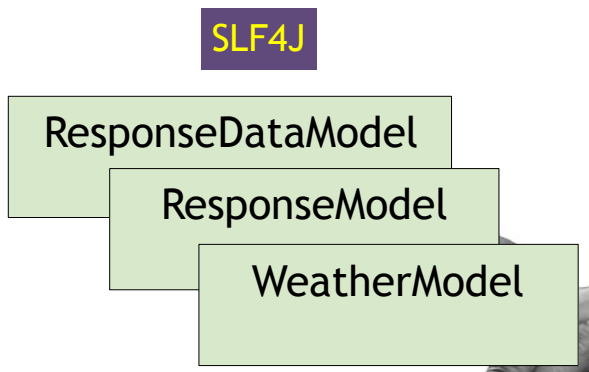
Java POWERED · Apache Tomcat · spring · MySQL

FIND YOUR PURPOSE

GRAND CANYON UNIVERSITY

IoT Services Application Logical Architecture





Spring Core

NOTE: All concrete implementations inserted by Spring IoC

Powered By

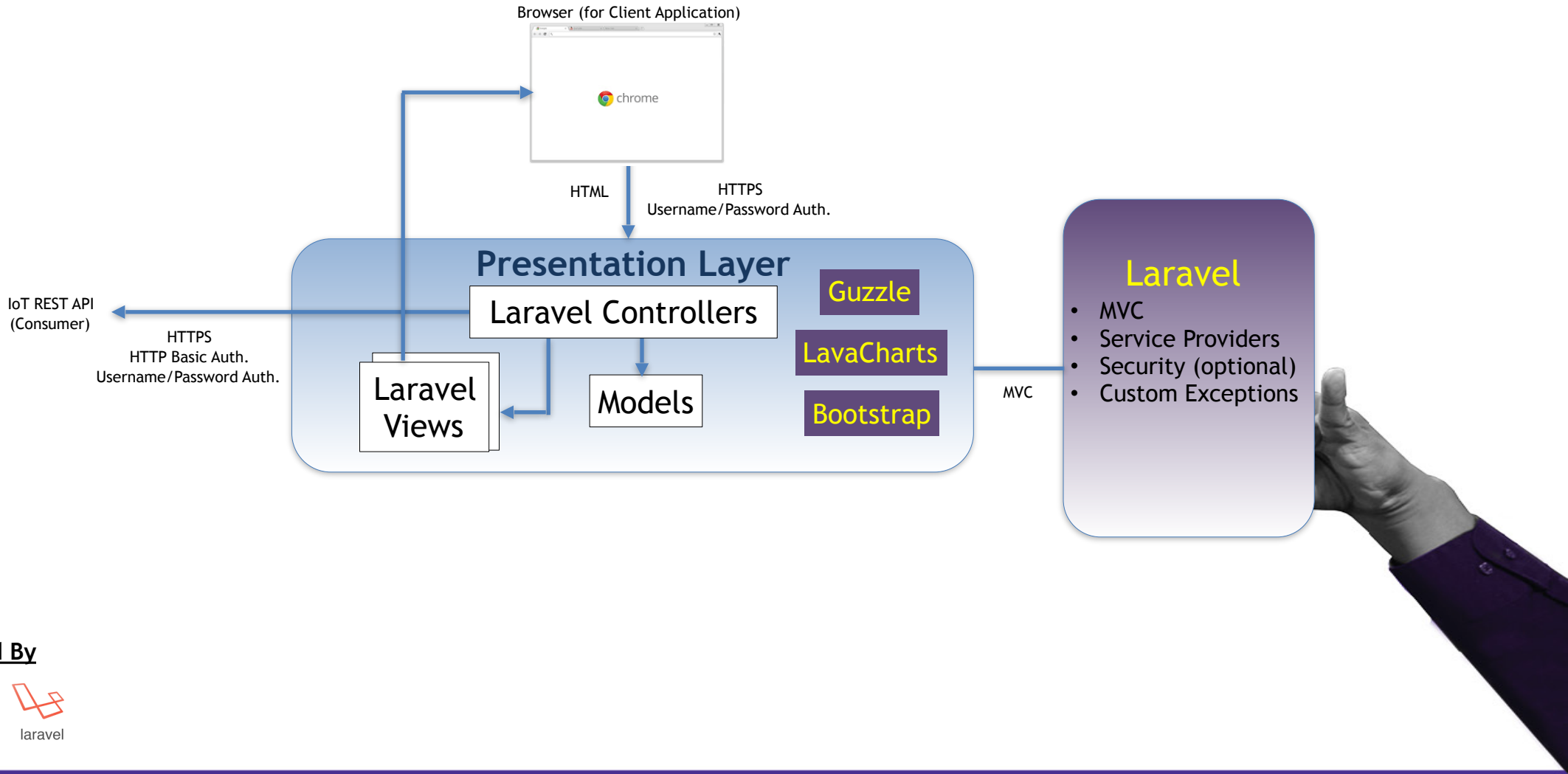
Java POWERED

Apache Tomcat

spring

MySQL

IoT Reporting Application Logical Architecture

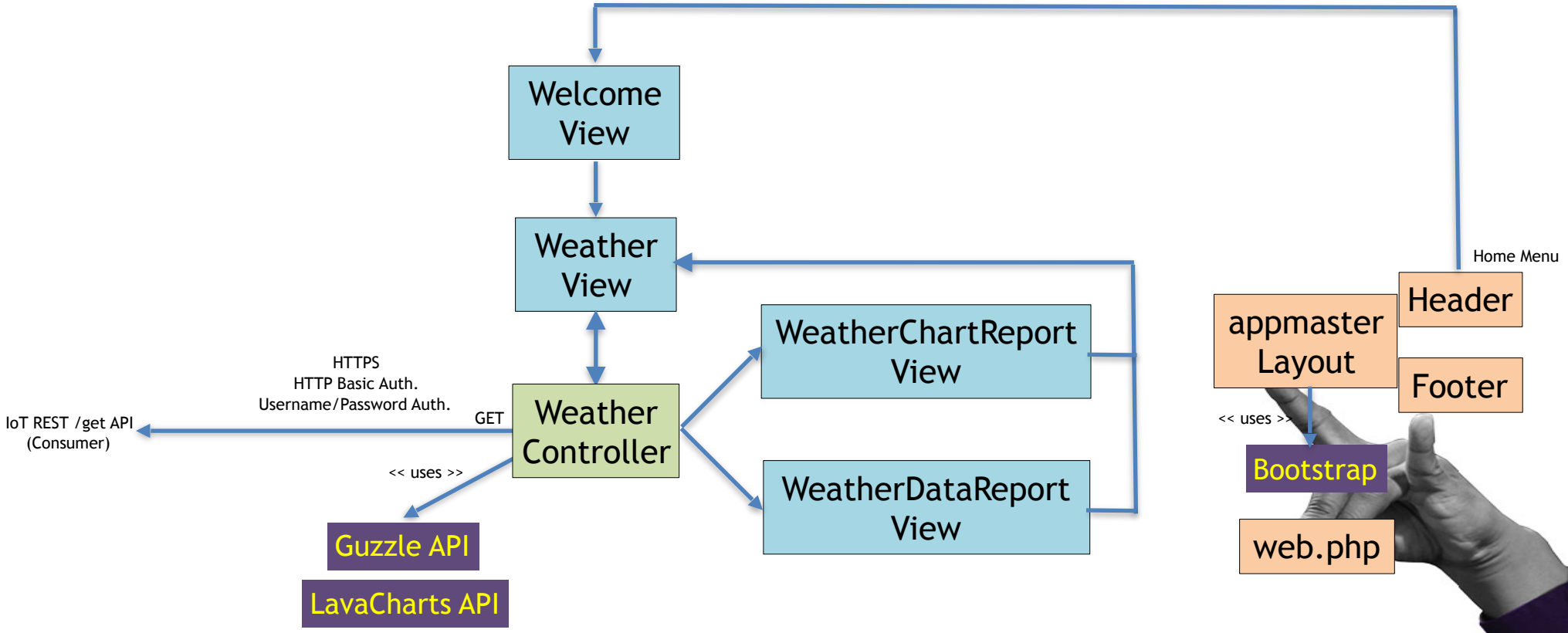


Powered By

php v7

laravel

IoT Reporting Application Logical Architecture

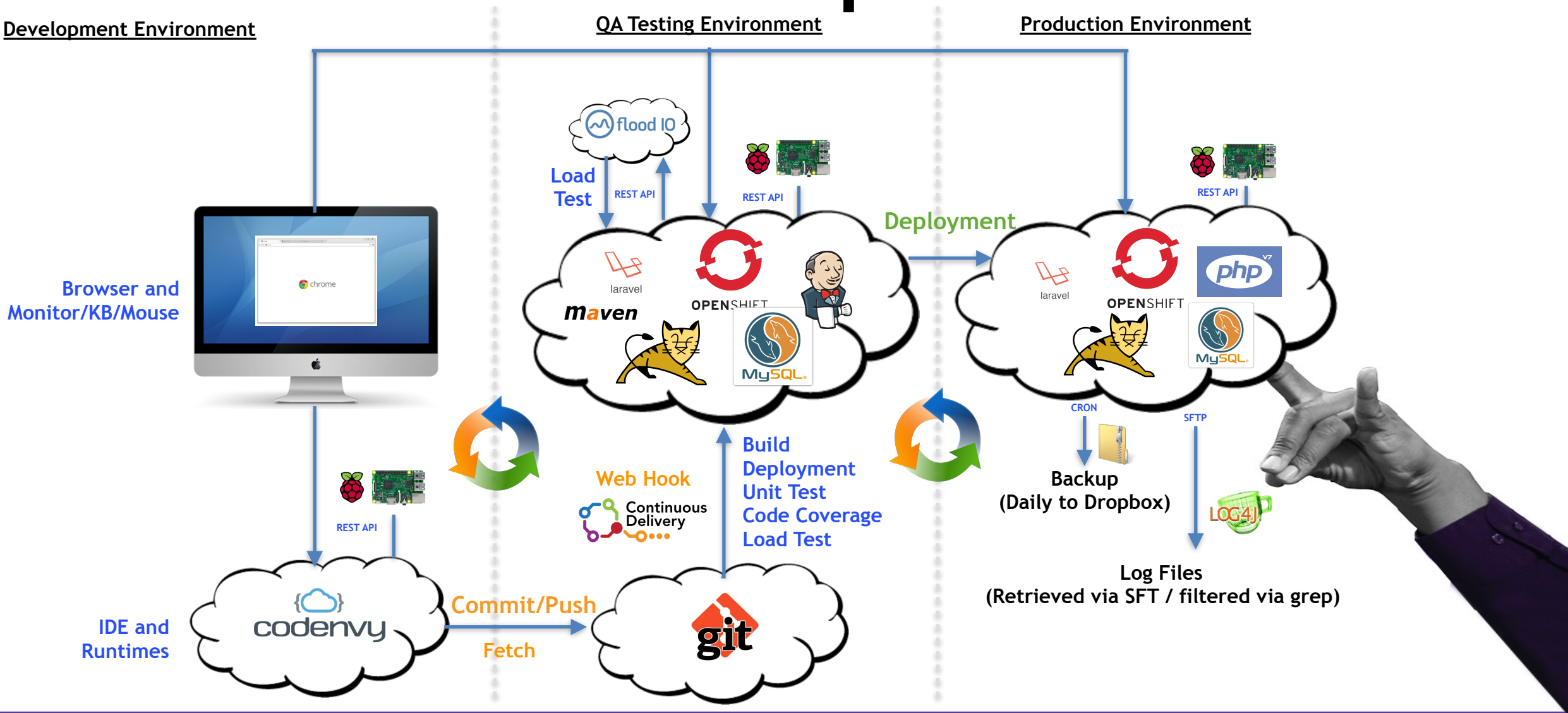


Powered By

php v7

laravel

Cloud Based Development Platform

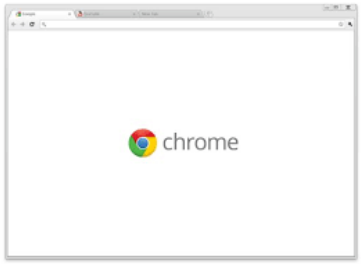


Physical Cloud Architecture

Client Tier

Raspberry Pi (for IoT Application)

Browser (for Client Application)

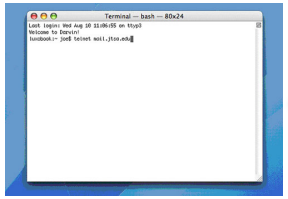


Server Monitoring

UptimeRobot

New Relic

SSH (for OpenShift Access)



Internet

HTTPS Port 443

HTTPS Port 443

SSH Port 22

Web-App Tier and Database Tier

Cloud PaaS Container

Apache Web Server

HTTP Port 8080

Tomcat

TCP Port 3306

MySQL

Cron Job

HA Enabled
2 CPU x 1G RAM

Powered By

OPENSHIFT Apache Tomcat MySQL

HTTPS Port 443

HTTPS Port 443

Cloud PaaS Container

Apache Web Server

HTTP Port 8080

PHP

HA Enabled
1 CPU x 1G RAM

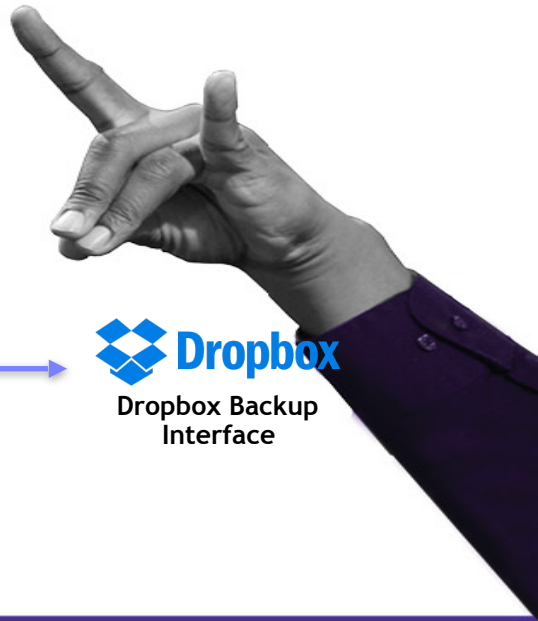
Powered By

OPENSHIFT php laravel

Internet

HTTPS Port 443

Integration Tier



Dropbox

Dropbox Backup Interface